Image Caption Generation

B.TECH PROJECT

Aanal Sonara (190070064)

Agenda

- Introduction
- Literature Survey
- Proposed Architecture
- Implementation
- Result and Analysis
- Current and Future Work

INTRODUCTION

Image caption generation is the task of automatically generating textual descriptions that accurately and concisely describe the contents of an image.

The process generally involves multi-modal learning where computer vision techniques are used to extract relevant features from the image and then using natural language processing methods to convert these features into a coherent and descriptive caption.

Image caption generation has many practical applications, including aiding the visually impaired, assisting in image retrieval etc

LITERATURE SURVEY

1) CLIPCLAP

The most relevant paper was CLIPCLAP which used a CLIP image encoder with GPT2 decoder through a linear mapping. This leads to results comparable to some existing work and even better as models get larger (due to generalisation of CLIP and GPT2). However, it faces the same limitations as CLIP i.e. it cannot work on counting objects or more complex tasks

2) IMAGE CAPTION GENERATION WITH HIGH LEVEL FEATURES

They have tried to mimic human cognitive capabilities. So first there is stimulus driven data like contrast, size. Second is convolution for feature extraction. They have used FasterRCNN or rather they go over patches of images to find out details.
Their model is capable of distinguishing single and multiple things and small things like "woman holding a glass of wine"
It cannot count. It's relatively old paper (2019), we can use Vision Transformers now

LITERATURE SURVEY

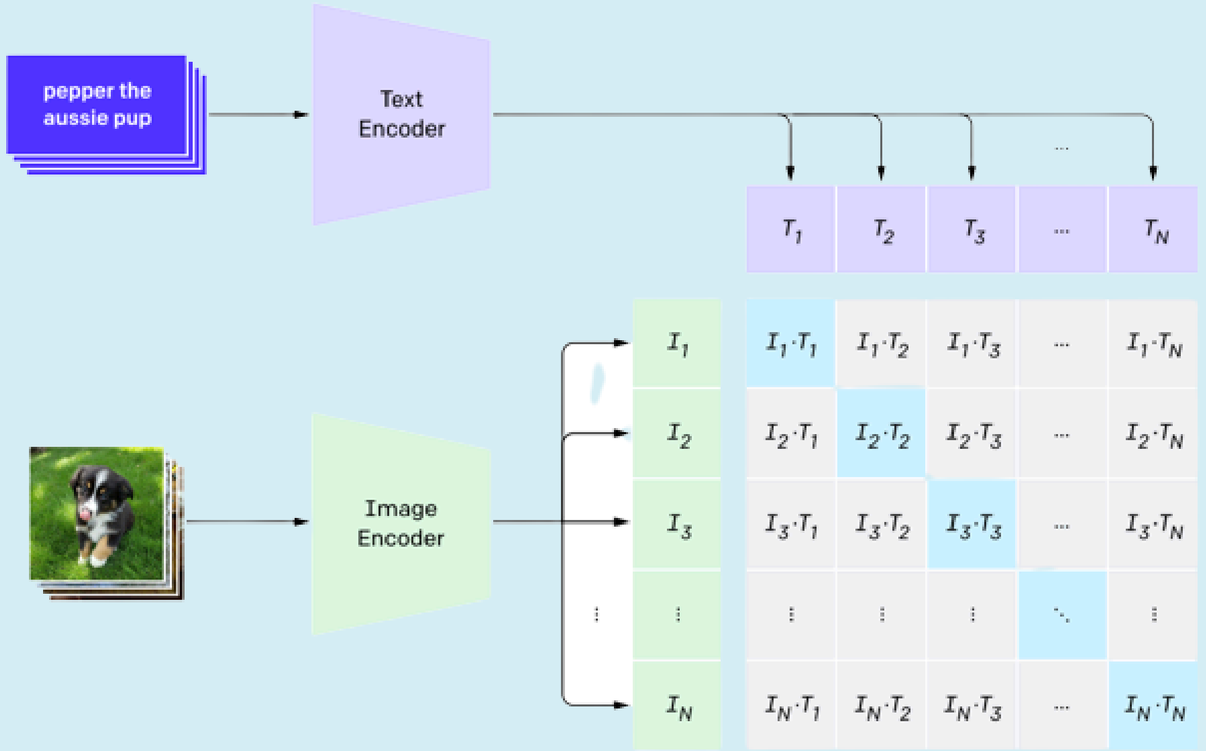
3) A HIERARCHICAL AND REGIONAL DEEP LEARNING ARCHITECTURE FOR IMAGE DESCRIPTION GENERATION

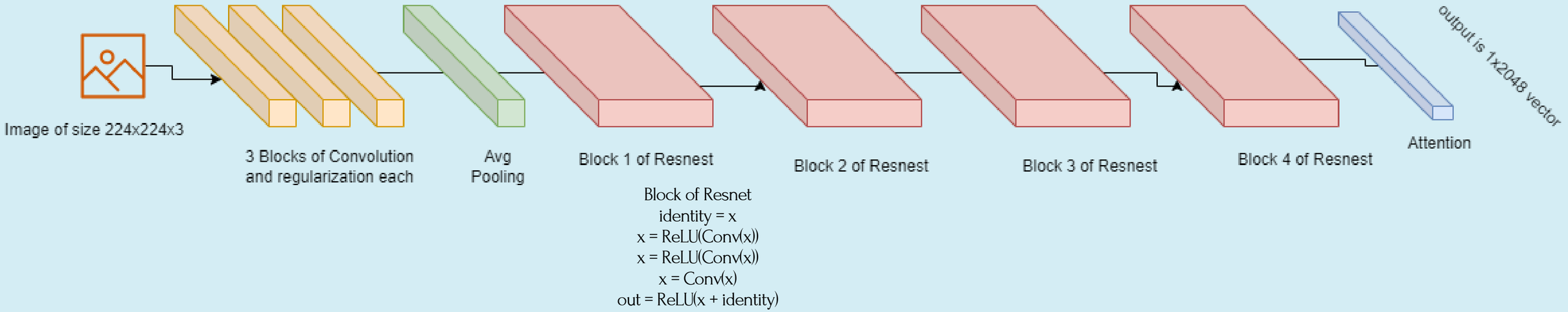
Heirarchy is in the sense of training (freezing some layers and training others). They have focused on description instead of mere caption generation. It is capable of accurately describing the scene and counting. They are using VGG16 backbone for image encoder. They have used window based labeling and then generating captions/descriptions

Their future suggests to use attention mechanism

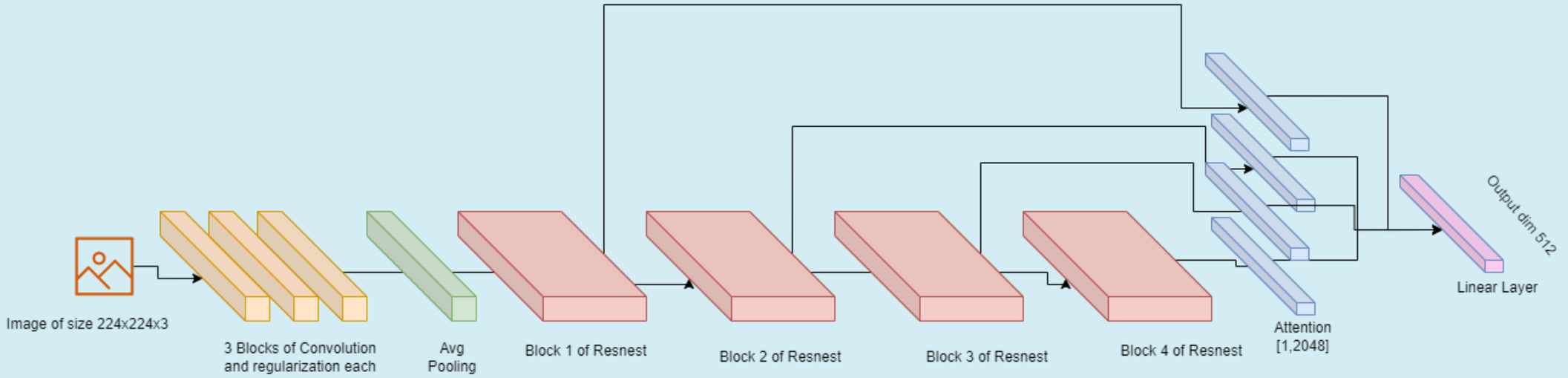
CLIP ARCHITECTURE

CLIP is used for image classification using multi-modal data or generating general captions like "this is an image of {class}"
CLIP is still limited when processing images with multiple objects and systematic tasks like which object is closer or farther





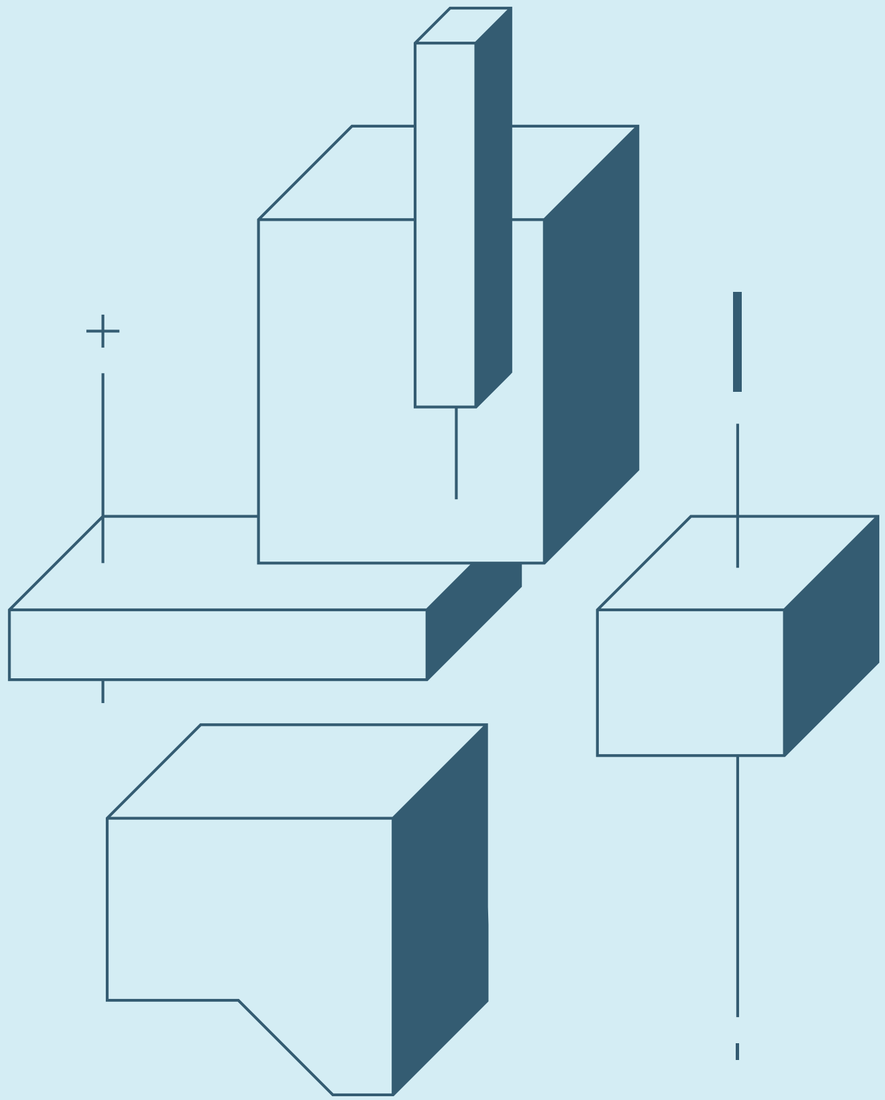
PROPOSED ARCHITECTURE



HEIRACHICAL FEATURE EXTRACTION

Using attention layer after each resnet block to increase information retrieval or extraction from network

CLIP tends to leave out small information such as boy standing next to train and it may omit out a cycle (boy is holding the cycle). This is probably because of the deep neural net architecture



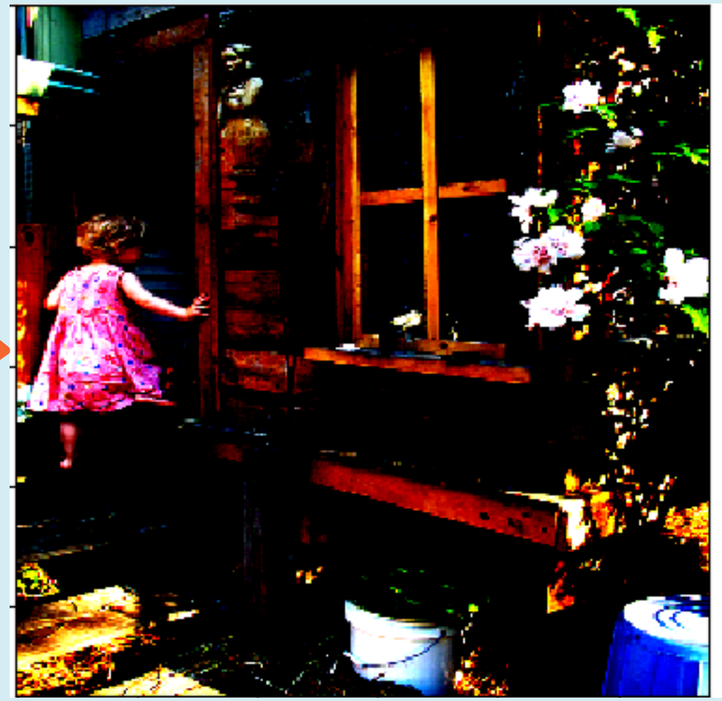
IMPLEMENTATION

CLIP BackBone - ResNet50 (small model due to limited computation time)
Dataset - Flickr8k (5 captions per image; approximately 8k images)
Loss - CrossEntropy
Metric - Bleu
Implemented on Google Colab and Kaggle
Major Issue - Making changes to OpenAI code and using it for training (clip does not have training details)

Preprocessing of Image

Image

Resizing to (224, 224, 3)



Normalize

MORE IMPLEMENTATION DETAILS

Data Augmentation

Image caption generation models are trained in a supervised setting
Thus if the training data is insufficient the models will also generate small captions or cannot describe image properly
Flick8k has 5 captions per image and each differ than the other not just in words (paraphrasing) but also in information this can cause confusion for the model
Thus I have OpenAI chatbot to summarise these captions such that no information is left out

1) A child in a pink dress is climbing up a set of stairs in an entry way .
2) A girl going into a wooden building .
3) A little girl climbing into a wooden playhouse .
4) A little girl climbing the stairs to her playhouse .
5) A little girl in a pink dress going into a wooden cabin .

Response - A young girl wearing a pink dress is seen climbing stairs in an entryway, entering a wooden building, climbing into a playhouse, and going into a wooden cabin.

MORE IMPLEMENTATION DETAILS

Data Augmentation (Fail)

1. A little girl covered in paint sits in front of a painted rainbow with her hands in a bowl .
2. A little girl is sitting in front of a large painted rainbow.
3. A small girl in the grass plays with fingerpaints in front of a white canvas with a rainbow on it .
4. There is a girl with pigtails sitting in front of a rainbow painting .
5. Young girl with pigtails painting outside in the grass .

A little girl is sitting in front of a painted rainbow with her hands in a bowl. Another small girl is playing with fingerpaints on a white canvas with a rainbow on it, while sitting in the grass. A third girl, who has pigtails, is painting outside on the grass. All three girls are depicted in scenes involving rainbow paintings.

MORE IMPLEMENTATION DETAILS

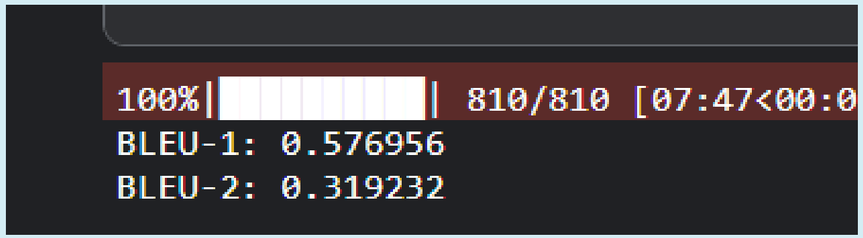
Dead - ends

There is almost zero working documentation on how to use CLIP as a custom model or train it and therefore simply loading the model via functions (not as a package) with pre-trained weights for training purpose is a challenge
Maximum input token for GPT2 is 1024
At first, I used image encoder output directly as an embedding input to GPT2 but it led to an error stating "self out of range" which was not resolved when I was trying to use it as an embedding (as a prompt)
GPT2 cannot be directly used with image feature vectors as they are not in the same "domain space" as words and therefore we use a simple linear mapping transformation
The output of this mapping is used as a prefix or prompt

RESULT AND ANALYSIS

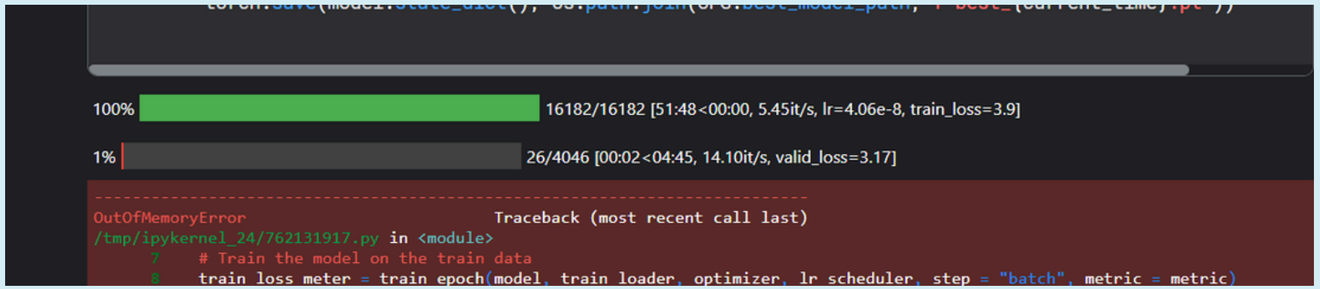
There were several issues faced in trying to implement GPT2 as a decoder with CLIP due to the lack of implementation available online; most people use huggingface for finetuning without making changes to arhcitecture
Thus, to create a baseline I have used a pre-trained CLIP image encoder (ResNet50) with an LSTM model. This experiment was completed end-to-end

CLIP Image encoder + LSTM



Bleu score is higher than using a deep CNN model as feature encoder and only trained on Flickr dataset

CLIP Image encoder + GPT2

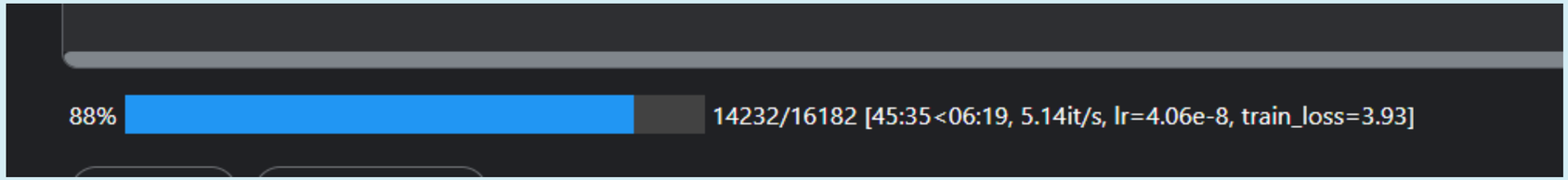


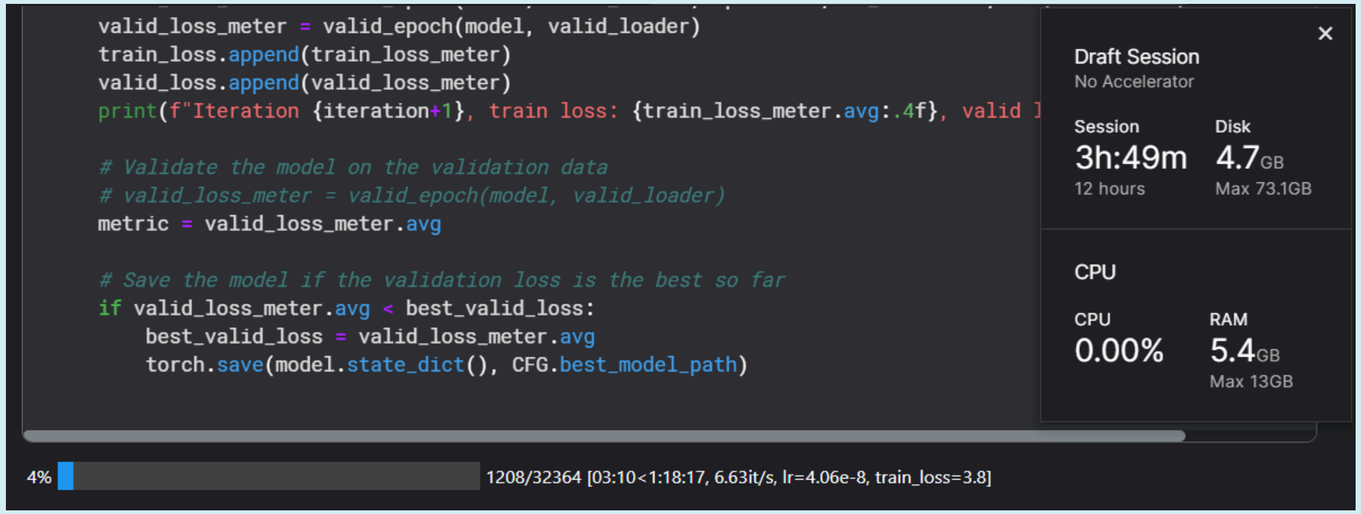
Train loss - 3.9
Valid loss (and bleu score) not obtained as it crashes before that

RESULT AND ANALYSIS

CLIPCLAP

Also testing against CLIPCLAP model to check for performance however facing an issue on Kaggle that model training freezes midway with no error. In the below image, session had been running for 4 hours. First it crashed at 88% when loss is 3.93 and second time after 4% itself





CURRENT WORK

I am in the process of wrapping up my project and completing the following things as we speak:

1) Training of CLIP custom encoder with GPT2
2) Completing the pipeline for getting BLUE/Rouge scores
3) Cleaning the code and removing dummy variables/functions
4) Making a detailed documentation (functions and variables, code control flow) for using CLIP code for any future purposes

FUTURE WORK

Some of the things that can be immediately tested as a continuation of this work

1) Test on augmented data
2) Train on larger datasets such as COCO or Conceptual Captions
3) Use this for downstream tasks such as Medical report generation
4) Use window based approach for feature vectors

ACKNOWLEDGEMENT

I would like to extend my gratitude to Prof. Biplab Banerjee for guiding me throughout project and helping me out of the deadends. His patience and support has been immensely helful to me. I also want to thank the DeepVC lab members who supported me with my various difficulties, be it, code debugging, discussing and researching about various approaches (which led to changing my approach midway) and their support thereafter till this point

THANKYOU